

Requested Patent: JP10260877A

Title:

PROTOCOL CONVERSION SYSTEM IN CLIENT SERVER SYSTEM, METHOD THEREFOR AND RECORDING MEDIUM PROGRAMMED AND RECORDED WITH THE METHOD ;

Abstracted Patent: JP10260877 ;

Publication Date: 1998-09-29 ;

Inventor(s): KUDOU NORIKIYO ;

Applicant(s): TOSHIBA CORP ;

Application Number: JP19970064077 19970318 ;

Priority Number(s): ;

IPC Classification: G06F12/00 ; G06F13/00 ; G06F13/00 ; H04L29/06 ;

Equivalents:

ABSTRACT:

PROBLEM TO BE SOLVED: To provide a protocol conversion system and method in a client server system, which accesses different file systems that exist in from a client to a server, and to construct a memory medium, in which the method is programmed and supplied as a middleware.

SOLUTION: An access request is issued to a file system 21 that exists at other nodes, and data is constructed according to a protocol that is preliminarily defined at a client server system 12, which exists at a self node 1 and delivers the request to a network. An access to the self file is executed by performing protocol conversion (protocol conversion mechanism 22) in order to make the access request access a file system 21 that is held by its own system 2 and it is converted into a protocol that is requested by the file system of a client and the execution result is delivered to the client who makes a request through a network 3.

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平10-260877

(43)公開日 平成10年(1998) 9月29日

(51)Int.Cl.⁶
G 0 6 F 12/00
13/00
H 0 4 L 29/06

識別記号
5 1 3
3 5 1
3 5 7

F I
G 0 6 F 12/00
13/00
H 0 4 L 13/00

5 1 3 J
3 5 1 B
3 5 7 Z
3 0 5 B

審査請求 未請求 請求項の数4 O L (全 5 頁)

(21)出願番号 特願平9-64077

(22)出願日 平成9年(1997) 3月18日

(71)出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72)発明者 工藤 則清

東京都青梅市末広町2丁目9番地 株式会

社東芝青梅工場内

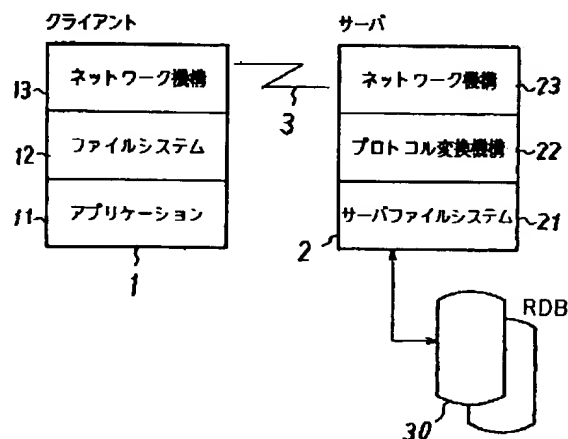
(74)代理人 弁理士 大胡 典夫 (外1名)

(54)【発明の名称】 クライアントサーバシステムにおけるプロトコル変換システム、ならびに方法、及び同方法がプログラムされ記録された記録媒体

(57)【要約】

【課題】 本発明は、クライアントからサーバに存在する異なったファイルシステムをアクセスする、クライアントサーバシステムにおけるプロトコル変換システムならびに方法、及び同方法がプログラムされミドルウェアとして供給されるメモリ媒体を提供ならびに構築することを主な特徴とする。

【解決手段】 他ノードに存在するファイルシステム21へのアクセス要求を発し、自ノード1に存在するクライアントファイルシステム12にてあらかじめ定められたプロトコルに従ってデータを構築してネットワークにその要求を引き渡し、そのアクセス要求を自システム2が持つファイルシステム21へアクセスするためにプロトコル変換(プロトコル変換機構22)を行って自ファイルへのアクセスを実行し、クライアントのファイルシステムが要求するプロトコルに変換してネットワーク3経由で要求のあったクライアントにその実行結果を引き渡す。



【特許請求の範囲】

【請求項1】 他ノードに存在するファイルシステムへのアクセス要求を発生し、自ノードに存在するクライアントファイルシステムにてあらかじめ定められたプロトコルに従ってデータを構築してネットワークにその要求を引き渡し、そのアクセス要求を自システムが持つファイルシステムへアクセスするためにプロトコル変換を行って自ファイルへのアクセスを実行し、クライアントのファイルシステムが要求するプロトコルに変換しネットワーク経由で要求のあったクライアントにその実行結果を引き渡すことを特徴とするクライアントサーバにおけるプロトコル変換方法。

【請求項2】 自他ファイルシステムに対するアクセス要求を発生するクライアントと、各ノードに存在するデータを上位層に対して統一的なアクセス手順で提供し、対象とするデータが他ノードに存在する場合でも下位層のネットワーク機構を介して同一インタフェースで提供するファイルシステムと、クライアントから送付されたファイルアクセスのプロトコルをサーバが有するファイルシステムをアクセスするための手続きに変換するプロトコル変換機構とを具備することを特徴とするクライアントサーバシステムにおけるプロトコル変換システム。

【請求項3】 サーバが持つファイルシステムは、オープンプラットフォームのリレーショナルデータベースであることを特徴とする請求項1記載のクライアントサーバシステムにおけるプロトコル変換システム。

【請求項4】 クライアントサーバシステムにおいて、他ノードに存在するファイルシステムへのアクセス要求を発生し、自ノードに存在するクライアントファイルシステムにてあらかじめ定められたプロトコルに従ってデータを構築してネットワークにその要求を引き渡し、そのアクセス要求を自システムが持つファイルシステムへアクセスするためにプロトコル変換を行って自ファイルへのアクセスを実行し、クライアントのファイルシステムが要求するプロトコルに変換しネットワーク経由で要求のあったクライアントにその実行結果を引き渡すようにプログラムし記録したことを特徴とする記録媒体。

【発明の詳細な説明】**【0001】**

【発明が属する技術分野】本発明は、クライアントサーバシステムにおけるプロトコル変換システム、ならびに方法、及び同方法がプログラムされ記録された記録媒体に関する。

【0002】

【従来の技術】ローカルのファイルシステムへのアクセスをアプリケーションプログラムの変更なしにネットワークワイドに拡張する機能が存在する。例えば、UNIXシステムにおけるNFS(Network File System)、PCシステムにおけるNetBIOS等である。

【0003】これらのシステムにおいて、アプリケーションプログラムが発行するファイルシステムへの要求は、決められたプロトコルに従いネットワークを介して相手機器に送付され、相手機器のファイルシステムへアクセスし、その結果をクライアントに返却する。

【0004】

【発明が解決しようとする課題】上述した従来例では、クライアントから発行された要求は、ローカルでの実行結果と同じ実行結果をそのままサーバで実行して得られることを要求するため、異なるファイルシステムあるいはアーキテクチャを有したサーバでは、本来そのサーバには存在しない、要求元のクライアントシステムと同一のファイルシステムをエミュレーションする手法により実現していた。従って、サーバが持つ特徴的なファイルシステムを有効に利用することができなかった。

【0005】本発明は上記問題を解決するためになされたものであり、クライアントから発行されたクライアントのファイルシステムをベースとしたプロトコルデータを、サーバが持つファイルシステムの機能にマッピングすることにより、旧来のファイルシステムを持つクライアントからサーバに存在する異なったファイルシステムをアクセスするための、クライアントサーバシステムにおけるプロトコル変換システムならびに方法、及び同方法がプログラムされミドルウェアとして供給される記録媒体を提供することを目的とする。

【0006】

【課題を解決するための手段】本発明は、クライアントサーバシステムにおいて、他ノードに存在するファイルシステムへのアクセス要求を発生し、自ノードに存在するクライアントファイルシステムにてあらかじめ定められたプロトコルに従ってデータを構築してネットワークにその要求を引き渡し、そのアクセス要求を自システムが持つファイルシステムへアクセスするためにプロトコル変換を行って自ファイルへのアクセスを実行し、クライアントのファイルシステムが要求するプロトコルに変換しネットワーク経由で要求のあったクライアントにその実行結果を引き渡すことを特徴とする。

【0007】また、本発明のプロトコル変換システムは、自他ファイルシステムに対するアクセス要求を発生するクライアントと、各ノードに存在するデータを上位層に対して統一的なアクセス手順で提供し、対象とするデータが他ノードに存在する場合でも下位層のネットワーク機構を介して同一インタフェースで提供するファイルシステムと、クライアントから送付されたファイルアクセスのプロトコルをサーバが有するファイルシステムをアクセスするための手続きに変換するプロトコル変換機構とを具備することを特徴とする。

【0008】更に、本発明の記録媒体は、クライアントサーバシステムにおいて、他ノードに存在するファイルシステムへのアクセス要求を発生し、自ノードに存在する

クライアントファイルシステムにてあらかじめ定められたプロトコルに従ってデータを構築してネットワークにその要求を引き渡し、そのアクセス要求を自システムが持つファイルシステムへアクセスするためにプロトコル変換を行って自ファイルへのアクセスを実行し、クライアントのファイルシステムが要求するプロトコルに変換しネットワーク経由で要求のあったクライアントにその実行結果を引き渡すようにプログラムされ記録され、ミドルウェアとして提供される。

【0009】このことにより、クライアントのアプリケーションプログラムからプログラムの変更を要せず異なるファイルシステムを持つサーバのデータをアクセスでき、サーバに存在するファイルシステムの特徴を有効に活用できる。

【0010】

【発明の実施の形態】図1は本発明の実施例を示すブロック図である。図において、1はクライアント、2はサーバであり、例えばTCP/IPプロトコルに従う通信回線3経由で接続されている。クライアント1は、アプリケーションプログラム11、クライアント独自のファイルシステム12、ネットワーク機構13から成る。

【0011】アプリケーションプログラム11は、ユーザインタフェースやアプリケーションロジックを有する部分であり、必要に応じ他ファイルシステムのサービスを要求する。ファイルシステム12は、各ノードに存在するデータを上位層（アプリケーション）に対して統一的なアクセス方法を提供する。対象とするデータが他ノード上に存在する場合でも下位層のネットワーク機能を介して同一インタフェースによって提供される。ネットワーク機構13は各ノードに割り付けられた異種コンピュータ間でデータの授受を行う、例えばTCP/IPプロトコルの実現手段である。

【0012】サーバ2は、サーバファイルシステム21、プロトコル変換機構22、ネットワーク機構23から成る。サーバ2はOSとしてオープンプラットフォームを搭載している。サーバ2が持つファイルシステム21としてRDB（リレーショナルデータベース）30が例示されている。プロトコル変換機構22は、クライアント1から送付されたファイルアクセスのプロトコルをサーバシステム2が有するファイルシステム21をアクセスするための手続きに変換するためのロジックであり、図2にその詳細が示されている。ネットワーク機構23は上述したネットワーク機構13と等価である。

【0013】以下、具体的動作について説明する。まず、アプリケーションプログラム11は、他ノードに存在するファイルへのアクセスをファイルシステム12へ要求する。ファイルシステム12は、他ノードに存在するファイルへのアクセスを認識し、決められたプロトコルに従ってデータの構築を行い、ネットワーク機構13にその要求を引き渡す。クライアント1のデータは、ネ

ットワーク機構13、通信回線3経由で目的とするサーバ2に送付されることになる。

【0014】プロトコル変換機構22は、クライアント1からのファイルシステムへのアクセス要求を自システムが有するファイルシステム21へアクセスするために図2に示すプロトコル変換を行い、自ファイルへのアクセスを行う。実行した結果は、再びこのプロトコル変換機構22でクライアント1のファイルシステム11が要求するプロトコルに変換し返却する。ネットワーク機構23を介してクライアント1に返却された実行結果は、以降の処理は通常のネットワークファイルアクセスと区別なく行われる。

【0015】図2は、プロトコル変換機構22による、データベースの接続からデータベース切断までの基本処理に対応する動作概要を示す図である。図中、図1と同じ番号の付されたブロックは図1のそれと同じとする。40は環境ファイルである。

【0016】環境ファイル40は、1つの環境ファイル内には複数のセクションを記述することができ、データベース毎接続情報を持つことができる。本発明実施例では、接続情報としてデータソース名（DSN）：SQL Server、ユーザIDがTPCARE、パスワードが****として記述される。続いてレコード取り出し形式が記述されている。データベースのテーブルはカラムの属性値を全て指定することになり、複数のデータレイアウト定義はREAD BINDでのみ指定が可能である。指定されたレコード形式でデータ変換が行われるが、対応するデータ形式として、Uはアンパック10進法、Xはデフォルト型、JはJIS漢字、Pはパック10進数を示す。最後に、COBOLで書かれたアプリケーションプログラム11がOPEN命令を発行したときに処理対象となるデータベースに実行するSQL文（対象とするデータベースのテーブルよりレコードを順読みする）を記述する。

【0017】プロトコル変換システム22は、クライアント1からファイルのアサイン（ASG）要求を受信することにより、ノード接続情報の指定環境ファイル30から環境情報を読み取る。ここで、ノード接続情報とは、ノード名TOKYO、環境ファイル名HONSH A. INI、セクション名ZAIKOをいい、SQL Connect関数を実行することにより環境情報の接続情報に従いデータベースの接続を行う。次に、環境ファイル30から対応するSQL文（SQL OPEN）を読み込みそのSQL文を実行（SQLExecDirect関数）してカーソルのオープンを行う。そして、SQLExttenderdFetch関数を実行して1つのレコードを取り出し、環境ファイル30に指定された出力レコード形式（BIND READ=*, *, *, *）に従いコード変換を行う。条件指定レコード取り出し時は、環境ファイルに記述されたSELECT文

(SQLREAD)を実行後レコードを取り出す。最後にDASG文で記述されるファイルのデアサイン要求を受け、プロトコル変換機構22は、SQLFreeStmt関数を実行することによりカーソルをクローズし、SQLDisconnect関数の実行によりデータベースの切断を実行する。

【0018】以上説明のように本発明は、クライアントから発行されたクライアントのファイルシステムをベースとしたプロトコルデータを、サーバが持つファイルシステムの機能にマッピングすることにより、旧来のファイルシステムを持つクライアントからサーバに存在する異なったファイルシステムをアクセスするためのプロトコル変換を提供するものであり、このことにより、クライアントのアプリケーションプログラムからプログラムの変更を要せず異なるファイルシステムを持つサーバのデータをアクセスでき、サーバに存在するファイルシステムの特徴を有効に活用できる。

【0019】

【発明の効果】以上説明のように本発明によれば以下に列挙する効果が得られる。

【0020】(1)旧来のアプリケーションプログラムからプログラムの変更を行うことなく、異なるファイル

システムを持つサーバ上に存在するデータにアクセスすることが可能となる。

【0021】(2)異なるファイルシステムを持つマシン間でのリアルタイムでのデータの授受が可能となり、結果として異なるアーキテクチャのマシンの共存共栄が容易に可能となる。

【0022】(3)旧来のアプリケーションプログラムでのローカルファイルシステムへアクセスするためのノウハウだけで、異なるアーキテクチャのファイルシステム上に存在するデータをアクセスできる。

【図面の簡単な説明】

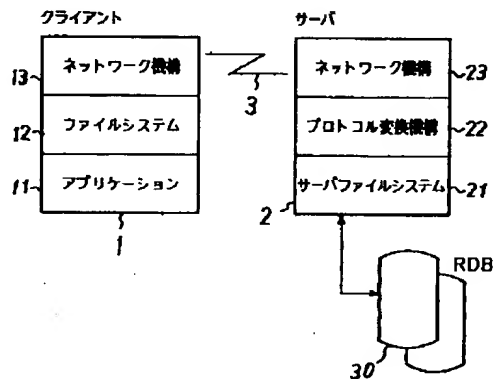
【図1】本発明実施例を示すブロック図。

【図2】本発明実施例によるデータベースの接続からデータベース切断に至る動作の流れを示す図。

【符号の説明】

1…クライアント、2…サーバ、3…通信回線、11…アプリケーションプログラム、12…クライアントファイルシステム、13、23…ネットワーク機構、21…サーバファイルシステム、22…プロトコル変換機構、30…リレーショナルデータベース、40…環境ファイル。

【図1】



【図2】

